What is claimed is:

1. A method for reducing a transaction time required to obtain a timeout values for remote transactions between nodes of a source bus and a destination bus, said method comprising:

(a) implementing a register table by a portal that contains a plurality of entries for storing respective remote timeout values from a local bus of a portal to a particular destination bus in a same net, wherein an Nth entry of the register table is corresponding to a bus ID of N;

(b) intercepting a TIMEOUT response message en route to a particular-addressed node by an exit portal if the TIMEOUT response message is addressed to the local bus of the portal, the TIMEOUT response message comprising remote timeout values;

(c) storing the remote timeout values contained in the TIMEOUT response message intercepted in step (b) in a corresponding entry of the plurality of entries in the register table implemented in step (a);

(d) forwarding the TIMEOUT response message intercepted in step (b) to the particular-addressed node;

(e) intercepting by a portal of a TIMEOUT request message from an initial requester, if the remote timeout values from the local bus of the portal to the destination bus to which the

intercepted TIMEOUT request message is addressed have been stored previously by step (c) in the register table recited in (a);

(f) synthesizing by the portal of a corresponding TIMEOUT response message having the remote timeout values for a remote transaction from the local bus of the portal to the destination bus where the intercepted TIMEOUT request from step (e) is addressed by one of:

(i) retrieving the remote timeout values from the register table if said initial requester of the TIMEOUT request message identified in step (e) is on the local bus of the portal; and

(ii)calculating the remote timeout values retrieved from the register table if said initial requester of the TIMEOUT request message identified in step (e) is not on the local bus of the portal, wherein a max_remote_payload value is the smaller of max_remote payload values in one of: (1) the intercepted TIMEOUT response message in step (b), and (2) the corresponding register table entry, and wherein remote timeout seconds, remote timeout cycles and hop count values in the intercepted TIMEOUT request message are added to the corresponding register table entry to the destination bus, respectively; and

(g) sending the TIMEOUT response message synthesized in step (f) to said initial requester of the TIMEOUT request message intercepted in step (e).

2. The method according to claim 1, wherein the storage area for the register table recited in step (a) comprises RAM.

3. The method according to claim 1, wherein least the source bus and destination bus comprise a serial bus connected by a bus bridge.

4. The method according to claim 3, wherein at least one intermediate bus is connected in a serial path between said source bus and said destination bus.

5. The method according to claim 1, wherein the source bus and destination bus recited in step (a) are connected via a bridged network.

6. The method according to claim 5, wherein said bridged network comprises a IEEE 1394 bridged network.

7. A method for reducing the transaction time to obtain a

total timeout values for remote transactions between nodes of a source bus and a destination bus, said method comprising the steps of:

(a) providing a source bus and a destination bus in a serial path having an intermediate bus connected between the source bus and destination bus via bridges;

(b) intercepting by a portal of said intermediate bus a TIMEOUT response message sent from an exit portal of the destination bus to a particular TIMEOUT requesting node on the source bus;

(c) storing first remote timeout values from the TIMEOUT response message intercepted in step (b) in a storage area of said portal of said intermediate bus and forwarding the TIMEOUT response message intercepted in step (b) to said particular TIMEOUT requesting node on the source bus;

(d) intercepting by said portal of said intermediate bus said intermediate bus a subsequent TIMEOUT request message from a subsequent requesting node of said plurality of nodes of said source bus to a node of said plurality of nodes of said destination bus;

(e) said portal of the intermediate bus calculating a remote timeout values from the source bus to the destination bus by adding the first timeout values stored in step (c) to second

remote timeout values between the source bus and the intermediate bus except for a max_remote payload value;

(f) synthesizing a TIMEOUT response message by said portal of said intermediate bus, said synthesized TIMEOUT RESPONSE including the total timeout values calculated in step (e); and

(g) forwarding said synthesized TIMEOUT response message to said subsequent requesting node of said source bus in step (d) that initiated the TIMEOUT request message.

8. The method according to claim 7, wherein the max_remote payload value is a smallest of max-remote payload values in the values intercepted in the message and a corresponding register table entry to the destination bus.

9. The method according to claim 8, wherein the source bus, intermediate bus, and destination bus are connected by a bridged network.

10. The method according to claim 7, wherein the storage area recited in step (c) is a RAM of a portal on said intermediate bus.

11. The method according to claim 7, further includes storing the timeout values of the TIMEOUT response message in a storage area of the source bus.

12. The method according to claim 11, wherein said storage area is a RAM of an entry portal.

13. The method according to claim 7, wherein said source bus, and said destination bus comprises a 1394 IEEE bridged network.

14. The method according to claim 12, wherein said source bus, said intermediate bus, and said destination bus comprise a 1394 IEEE bridged network.

15. The method according to claim 13, wherein the network includes a plurality of buses serial connected between the source bus and the destination bus.

16. A serial bus bridged network having a reduced-response timeout management system, comprising

a source bus having a first portal with a register table;

at least one intermediate bus having a bridge comprising a second portal including a register table;

a destination bus having an exit portal;

a plurality of bus bridges which serially connects said source bus, said at least one intermediate bus and said destination bus;

said first portal of the source bus including means for receiving a TIMEOUT request from a node attached thereto;

said exit portal of said destination bus including means for receiving a TIMEOUT request message and for sending a TIMEOUT response message having timeout values included therein;

intercepting and storing means for intercepting by the bridge portal of said intermediate bus the TIMEOUT response message sent by the exit portal of the destination bus, and for storing timeout values of the TIMEOUT response message from the exit portal of said destination bus in the register table of the bridge portal of the intermediate bus;

means for intercepting and synthesizing, in which said second portal of said intermediate bus intercepts a TIMEOUT request from a node on said source bus, and when an ID of the destination bus in the TIMEOUT request matches an ID of said destination bus having its timeout values stored in the register table of the second portal of the intermediate bus, the second portal synthesizing a TIMEOUT response message comprising a total timeout values by adding the timeout values between said destination bus and said intermediate bus and the timeout values between said source bus and said intermediate bus, except for

max_remote payload values, which is a smaller of the intercepted TIMEOUT request message and corresponding table entry,

means for transmitting the synthesized TIMEOUT response message having the total timeout values sent to the requesting node on said source bus.

17. The apparatus according to claim 16, wherein said means for storage comprises RAM.